

การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือระหว่าง IoT Cloud Platform Application และ End-to-End Application กรณีศึกษาเครื่องให้อาหารปลา

โสภณวิษณุ เขียวคำจิ้น¹

ชัยพร เชมะภาคพันธ์²

บทคัดย่อ

วัตถุประสงค์ของสารนิพนธ์ฉบับนี้เป็นการศึกษาออกแบบและเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือระหว่าง IoT Cloud Platform Application และ End-to-End Application โดยกรณีศึกษาผ่านเครื่องให้อาหารปลา ปัจจุบันแอปพลิเคชันโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือมีการพัฒนาและเกิดขึ้นจำนวนมาก สามารถพัฒนาได้ทั้งรูปแบบ IOT Cloud Platform Application และ End-to-End Application จึงจำเป็นต้องศึกษาและทดสอบว่าแอปพลิเคชันโปรแกรมประยุกต์รูปแบบไหนมีคุณสมบัติและประสิทธิภาพเหมาะสมต่อการนำไปใช้งานได้

โดยงานวิจัยส่วนที่เป็น IoT Cloud Platform Application จะใช้แอปพลิเคชัน Blynk เป็นตัวศึกษาออกแบบและทดสอบ และในส่วน End-to-End Application จะใช้ แอปพลิเคชัน Inventor 2 เป็นตัวศึกษาออกแบบและทดสอบ โดยภาพรวมของงานวิจัยทั้งหมดจะเป็นการออกแบบและศึกษาและเปรียบเทียบประสิทธิภาพของโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือระหว่าง IOT Cloud Platform Application และ End-to-End Application โดยทดสอบประสิทธิภาพด้าน User Interface ประสิทธิภาพด้านเวลา และประสิทธิภาพด้านค่าใช้จ่าย โดยจะทดสอบผ่านเครื่องให้อาหารปลา

จากผลการทดสอบพบว่าการพัฒนาแอปพลิเคชันบน IoT Cloud Platform Application ให้ประสิทธิภาพการทำงานที่ช้ากว่าเล็กน้อย แต่มีขั้นตอนการพัฒนาและการนำไปใช้งานที่สะดวกกว่าเมื่อเทียบกับ End-to-End Application

Abstract

¹ นักศึกษาด้านวิศวกรรมคอมพิวเตอร์และโทรคมนาคม วิทยาลัยนวัตกรรมการเรียนการสอนเทคโนโลยีและวิศวกรรม มหาวิทยาลัยบูรพา

² อาจารย์ที่ปรึกษา

The objective of this thematic research is to design, implement and compare the development of mobile-based IoT application between IoT Cloud Platform Application and End-to-End Application in case of fish feeder. Currently, mobile-based IoT application can be developed by using both IoT Cloud Platform Application and End-to-End Application. Thus, the study of which one is suitable and provides the acceptable performance to develop a mobile-based IoT application .

Based on IoT Cloud Platform Application, Blynk is chosen for use in design and implement. Inventor2 is chosen for use in design and implement for End-to-End Application. The testing results include the performances, user interface design and cost of implementation.

From the studied results, it can be found that IoT Cloud Platform Application provides the worse performance than End-to-End Application. However, IoT Cloud Platform Application has more familiar in design, implementation and applying for use in comparison with End-to-End Application.

บทนำ

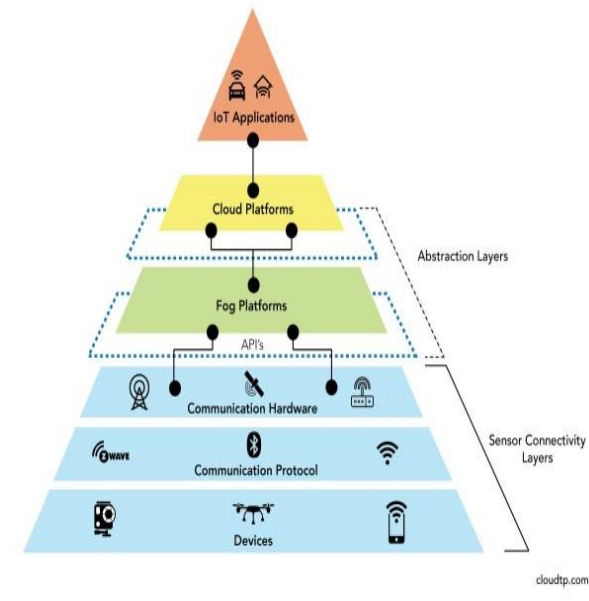
ในปัจจุบันอินเทอร์เน็ตสรรพสิ่ง(Iot) เริ่มมีบทบาทความสำคัญต่อชีวิตประจำวันมนุษย์อย่างมาก จึงทำให้เกิดการพัฒนาและออกแบบแอปพลิเคชันออกมาจำนวนมากที่ใช้ในการเชื่อมต่อและสั่งการผ่านอุปกรณ์(Iot) จึงทำให้เป็นที่นิยมและแพร่หลายอย่างรวดเร็วในปัจจุบัน จึงจำเป็นต้องหาแอปพลิเคชันที่มีประสิทธิภาพในการใช้งานและการทำงานที่ดีที่สุด เพื่อให้บรรลุตามวัตถุประสงค์และความต้องการของผู้ใช้งานต่อไป

2.1 ทฤษฎีที่เกี่ยวข้อง

หัวข้อนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องของงานวิจัยนี้ โดยแบ่งออกเป็น 2 ส่วน ได้แก่ ส่วนที่ 1 คือ IoT Cloud Platform Application จะใช้แอปพลิเคชัน Blynk เป็นตัวศึกษาออกแบบและทดสอบ และในส่วนที่ 2 คือ End-to-End Application จะใช้แอปพลิเคชัน Inventor 2 เป็นตัวศึกษาออกแบบและทดสอบ โดยภาพรวมของงานวิจัยทั้งหมดจะเป็นการออกแบบและศึกษาทดสอบและเปรียบเทียบประสิทธิภาพของโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือระหว่าง IoT Cloud Platform Application และ End-to-End Application โดยจะทำการทดสอบและเปรียบเทียบประสิทธิภาพดังต่อไปนี้ ประสิทธิภาพด้าน User Interface ประสิทธิภาพด้านเวลา และ ประสิทธิภาพด้านค่าใช้จ่าย

2.1.1 องค์ประกอบสำคัญในการเชื่อมต่อ Internet of Things เข้ากับ Cloud

[1] การมาของ Internet of Things (IoT) นี้เป็นหนึ่งในสิ่งที่ทำให้วงการ IT มีการเปลี่ยนแปลงมากที่สุดอย่างหนึ่งในปัจจุบัน ไม่ว่าจะเป็นระบบเครือข่าย, Sensor, อุปกรณ์ประมวลผล, ซอฟต์แวร์จัดเก็บข้อมูล, ซอฟต์แวร์วิเคราะห์ข้อมูล หรือแม้แต่ระบบ Cloud เองก็ตาม และนอกจากการเปลี่ยนแปลงแล้ว Layer ใหม่ที่เกิดขึ้นมาในระบบ Ecosystem นี้อย่าง Fog Platform ก็เป็นอีกหนึ่งสิ่งที่น่าสนใจด้าน Internet of Things



รูปที่ 2.1 ภาพตัวอย่าง การเชื่อมต่อ Internet of Things เข้ากับCloud

Fog Platform: PaaS สำหรับ Internet of Things

ในการพัฒนา Internet of Things Application ใดๆ นั้น ความซับซ้อนที่จะเพิ่มขึ้นมาจากในอดีตก็คือชั้นของการรับข้อมูลจาก Sensor ต่างๆ ที่มีความหลากหลายมาประมวลผล ซึ่งการพัฒนาระบบให้สามารถรองรับ Behavior ต่างๆ ของ Sensor ที่มีความแตกต่างกันทั้งในเชิงของผู้ผลิต, Protocol, ข้อมูลที่รับส่ง และวิธีการเข้ารหัสนั้นก็ถือเป็นงานที่ยากลำบากเป็นอย่างมากอย่างแน่นอน ซึ่งในปัจจุบันนี้ที่เรายังไม่เห็นความซับซ้อนตรงนี้มากนัก เพราะ IoT ตอนนี้ยังอยู่ในระยะเริ่มต้นที่ Sensor ส่วนใหญ่ในแต่ละ Application ยังเป็น Sensor กลุ่มเดียวกันอยู่ ซึ่งในอนาคตภาพเหล่านี้ก็จะค่อยๆ หายไป

นอกจากนี้ การนำ IoT ไปประยุกต์ใช้ในแต่ละกลุ่มอุตสาหกรรมนั้นก็มีความแตกต่างกันในเชิงกฎหมายและข้อบังคับหรือ Compliance ที่องค์กรต้องเผชิญ ระดับความปลอดภัยที่ต้องทำการควบคุม หรือ ความสำคัญของข้อมูลที่ต้องจัดเก็บและวิเคราะห์นั้นก็แตกต่างกันออกไป ทำให้การพัฒนา IoT Application แต่ละครั้งนั้น ไม่ใช่เรื่องง่ายและนำมา Reuse ได้ค่อนข้างยาก

Fog Platform คือระบบที่เกิดขึ้นมาเพื่อแก้ไขสองปัญหาหลักๆ เหล่านี้ โดย Fog Platform นี้จะอยู่กึ่งกลางระหว่างชั้นของ Cloud Platform ที่ทำหน้าที่จัดเก็บข้อมูล, ประมวลผล และแสดงผลข้อมูลให้กับผู้ใช้งาน กับชั้นของ Sensor ที่ส่งข้อมูลต่างๆ ผ่านมายังระบบเครือข่ายนั่นเอง

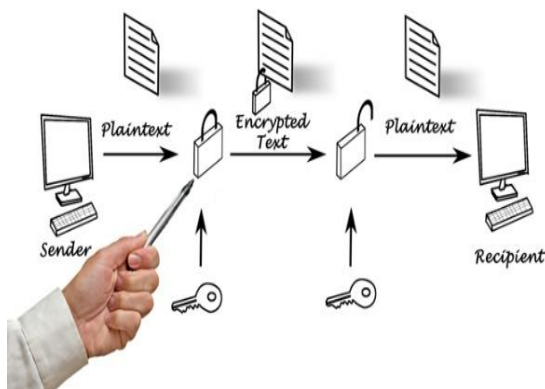
เรียกได้ว่า Fog Platform นี้คือ Platform-as-a-Service (Paas) สำหรับการทำให้ Internet of Things โดยเฉพาะ เพื่อให้การเชื่อมต่อและการควบคุม IoT Sensor สามารถทำได้ง่ายขึ้น รวมถึงการประมวลผลข้อมูลขั้นต้นเองก็จะทำได้อย่างรวดเร็วและง่ายดาย ตรงตามความต้องการของข้อกำหนดต่างๆ ในแต่ละอุตสาหกรรม ทำให้เหล่านักพัฒนาสามารถ Focus กับการพัฒนา Application ที่นำข้อมูลจาก Sensor และ

สั่งการกลับไปยัง Sensor ต่างๆ ได้อย่างมีคุณภาพมากยิ่งขึ้นนั่นเอง และ Fog Platform จะพบปัญหาแบบเดียวกับ Cloud Platform ในปัจจุบันนี้เราเริ่มเจอกับการต้องใช้งาน Cloud Platform จากหลายๆ ผู้ให้บริการเพื่อนำมาสร้างเป็น Application เดียว และทำให้การบริหารจัดการระบบ Cloud หลายๆ ระบบนี้เป็นอีกปัญหาหนึ่ง ซึ่ง Fog Platform เองก็จะต้องพบกับปัญหาลักษณะนี้เช่นกัน และผู้ใช้งานก็ต้องเตรียมรับมือกับการบริหารจัดการหลายๆ Fog Platform นี้พร้อมๆ กันให้ได้ และ Internet of Things จะทำให้ Cloud เติบโตอย่างรวดเร็วยิ่งกว่าเดิมด้วยปริมาณของข้อมูลที่มาหาศาล และการมาของ Fog Platform นี้ก็จะเป็นตัวเร่งให้ Cloud เติบโตยิ่งขึ้นไปอีกสำหรับการจัดการ IoT Device จำนวนมหาศาลที่กำลังจะเกิดขึ้นในตลาดในอนาคตนี้ อย่างไรก็ตาม Fog Platform นี้ก็จะกลายเป็นอีกเทคโนโลยีที่ทุกๆ องค์กรไม่สามารถหลีกเลี่ยงได้เช่นเดียวกับ Cloud เพราะการที่จะนำเทคโนโลยี IoT มาใช้งานให้ได้อย่างรวดเร็วและปลอดภัยนั้น การใช้ Fog Platform จึงถือเป็นหนทางที่จะทำให้ขีดความสามารถในการแข่งขันสูงยิ่งขึ้น

2.1.2 ระบบ E2EE หรือ End-to-End Encryption ระบบเข้ารหัสตั้งแต่ต้นทางถึงปลายทาง

[2] End-to-End Encryption หรือ E2EE คือการเข้ารหัสผ่านระหว่างคู่สนทนาสองคนตั้งแต่ต้นทางถึงปลายทาง ที่มีผู้ให้บริการหลายรายใช้โฆษณากันว่าบริการส่งข้อมูลของตนมีความปลอดภัย จากการโดนดักฟัง หรือการสอดส่องข้อมูล ระบบ E2EE หรือ End-to-End Encryption มีวิธีการทำงานอย่างไร ยกตัวอย่าง เช่น ให้มีคู่สนทนาเป็นนาย A และนาย B เมื่อนาย A ส่งข้อความหานาย B โดยเปิดระบบ E2EE ข้อความที่ออกไปหานาย B จะถูกแปลงเป็นรหัสระหว่างทาง โดยมีถนน คือ อินเทอร์เน็ต ระหว่างที่ข้อความถูกส่งนั้น หากมีผู้ไม่ประสงค์ดีต้องการดูข้อความก็จะแสดงเป็นแค่รหัสเท่านั้น ไม่สามารถเห็นข้อความที่นาย A ส่งถึงนาย B ได้แบบตรงๆ

สรุป คือ การเข้ารหัสตั้งแต่ต้นทางถึงปลายทาง ช่วยให้เรามั่นใจได้ว่าข้อความของเราจะถูกเปลี่ยนเป็นความลับ โดยผู้ส่งดั้งเดิม และสามารถถอดรหัสได้โดยผู้รับปลายทางเท่านั้น รูปแบบอื่นๆ ในการเข้ารหัสอาจขึ้นอยู่กับวิธีการเข้ารหัสที่ดำเนินการโดยบุคคลภายนอก โดยทั่วไปแล้วการเข้ารหัสแบบต้นทางถึงปลายทางจัดว่าเป็นวิธีที่ปลอดภัยกว่าแบบอื่น เนื่องจากสามารถลดจำนวนของผู้ที่สามารถเข้าแทรกแซงหรือแฮ็กการเข้ารหัสได้



รูปที่ 2.2 ภาพตัวอย่าง End-to-End Encryption

2.2.3 ความแตกต่างระหว่าง IoT Cloud Platform Application และ End-to-End Application

IoT Cloud Platform Application จะมี Server ที่ทำหน้าที่เป็นตัวกลางในการสื่อสารระหว่าง แอปพลิเคชันบนมือถือกับเครื่องให้อาหารปลา ทำให้สามารถนำเครื่องให้อาหารปลาไปใช้งานที่ไหนก็ได้ เนื่องจากอุปกรณ์ทั้ง 2 ฟังก์ชันทำงานเป็น Client จึงสะดวกต่อการใช้งาน

แต่ในส่วน End-to-End Application จะไม่มี Server เป็นตัวกลางในการสื่อสาร จึงทำให้มีข้อจำกัดในการใช้งานนอกพื้นที่ ก็จะต้อง Fix Ip ให้อุปกรณ์ทั้ง 2 ฟังก์ชันสามารถมองเห็นกันได้จากภายนอกจึงจะสามารถติดต่อสื่อสารกันได้จึงทำให้ยุ่งยากกว่า IoT Cloud Platform Application และในส่วนความปลอดภัยในการรับส่งข้อมูล IoT Cloud Platform Application จะมีการเข้ารหัสรับส่งข้อมูลที่ปลอดภัยกว่า End-to-End Application อีกด้วย

2.2 งานวิจัยที่เกี่ยวข้อง

[3] สุชาดา พลาชัยภิมย์ศิริล (บทคัดย่อ : 2554) ได้วิจัยเรื่อง แนวโน้มการใช้แอปพลิเคชัน โดยแนวโน้มการใช้งาน Mobile Device อย่างสมาร์ตโฟนเพิ่มขึ้นอย่างก้าวกระโดดในช่วงไม่กี่ปีที่ผ่านมา ซึ่งเป็นผลมาจากการพัฒนา Mobile Applications และเทคโนโลยี ของตัวเครื่องโทรศัพท์จากค่าย ผู้ผลิตโทรศัพท์ โดยเฉพาะการพัฒนาต่อยอดแอปพลิเคชันบนอุปกรณ์เคลื่อนที่ของบริษัทต่างๆ ที่ แข่งขันกันเพื่อชิงความเป็นหนึ่งในตลาดด้าน Mobile Application ซึ่งการพัฒนาแอปพลิเคชันแบ่งเป็นการ พัฒนาแอปพลิเคชันระบบ (Operation System) และแอปพลิเคชันซอฟต์แวร์ที่ ตอบสนองการใช้งานบนอุปกรณ์ และด้วยแอปพลิเคชันที่เพิ่มขึ้นและมีประสิทธิภาพมากขึ้น

[4] พิชิต บุญครอง* และ พิศณุ บุรมศรี ได้วิจัยเรื่องการพัฒนาแอปพลิเคชันเพื่อการพยากรณ์อนุกรมเวลาโดยใช้ MIT App Inventor MIT App Inventor-Based Application for Time Series Forecasting สาขาวิชาสารสนเทศการลงทุน วิทยาลัยเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยรังสิต โดยงานวิจัยนี้นำเสนอหลักเบื้องต้นในการพัฒนาแอปพลิเคชันสำหรับการพยากรณ์อนุกรมเวลาโดยใช้ MIT App Inventor แอปพลิเคชันจะประมวลผลข้อมูลอนุกรมเวลาวิชุดเพื่อการพยากรณ์เชิงปริมาณพื้นฐาน 8 วิธี โครงการงานวิจัยนี้บูรณาการระหว่าง ระเบียบวิธีพยากรณ์และการเขียนแอปพลิเคชันผ่าน Google App Inventor Servers, MIT App Inventor และ สมาร์ตโฟน ระบบแอนดรอยด์ การพยากรณ์เชิงปริมาณของอนุกรมเวลาควรใช้ตัวแบบที่มีความซับซ้อนน้อยที่สุดและให้ความคลาดเคลื่อนน้อย ที่สุดเป็นที่คาดหวังว่าแอปพลิเคชันที่พัฒนาขึ้นจะอำนวยความสะดวกและเป็นประโยชน์ต่อการตัดสินใจที่เกี่ยวข้องกับอนุกรมเวลา ในชีวิตประจำวันได้

[5] จามจุรี กุลยอด1 , ศิลป์ณรงค์ ฉวีพัฒน์2 ได้วิจัย เรื่องต้นแบบระบบควบคุมการเปิด-ปิดไฟผ่านแอปพลิเคชัน บนระบบปฏิบัติการแอนดรอยด์ Wireless Light Switch Prototype with an Android Application สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกำแพงเพชร โดยงานวิจัยนี้ได้พัฒนาต้นแบบ ควบคุมการปิดเปิดไฟโดยใช้อุปกรณ์โน้ด ESP8266/NodeMCU โดยต่อผ่านชุดควบคุมสวิตซ์ไฟฟ้า 220 โวลต์ ผลจากการทดสอบ สามารถที่จะควบคุมการเปิด-ปิดไฟ ผ่านแอปพลิเคชันบนอุปกรณ์ที่รองรับระบบปฏิบัติการ แอนดรอยด์นอกจากนี้ยังสามารถตรวจสอบสถานะการทำงานของอุปกรณ์ผ่าน

แอปพลิเคชันดังกล่าวได้ว่าความสำคัญ: ESP8266 / NodeMCU / ระบบปฏิบัติการ แอนดรอยด์/ ส่วนควบคุม เปิด-ปิดไฟ

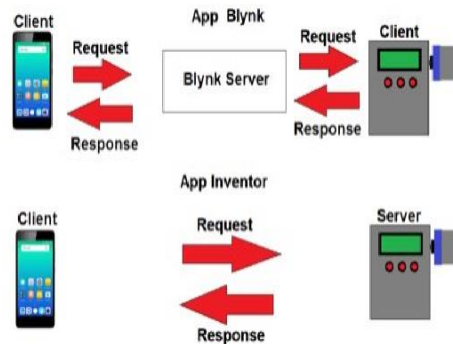
[6] สุจินต์ ณ นคร ได้วิจัยเรื่องระบบควบคุมผักสลัดอัจฉริยะ (Smart Salad Planting) สาขาวิชา วิศวกรรมวิทยาลัยนวัตกรรมการเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตโดยงานวิจัยนี้ ได้ทำการประยุกต์ใช้ IoT เพื่อมาช่วยจัดการระบบควบคุมปลูกผักสลัดอัจฉริยะ โดยอาศัยหลักทฤษฎีของ ระบบไฮโดรโปนิกส์ที่สามารถควบคุมการเพาะปลูกโดยที่สามารถควบคุมได้จากระยะไกลผ่าน โทรศัพท์เคลื่อนที่ เพื่อการใช้งานที่สะดวกสบาย พร้อมทั้งมีระบบจัดเก็บข้อมูล เพื่อติดตามผลได้ในระดับ ต่อไป โดยการนำอุปกรณ์เช่นเซอร์ต่าง ๆ มาติดตั้ง พร้อมทั้งการทำงานในระบบย่อยส่วนอื่น ๆ ในการจัดการ ระบบโดยระบบควบคุมการปลูกผักสลัดอัจฉริยะ สามารถใช้เพาะปลูกผักสลัดและการเพาะปลูกพืชในระบบ ไฮโดรโปนิกส์ต่างๆ ยังนำไปประยุกต์ใช้กับพืชผักชนิดอื่น ๆ ได้หลากหลายในระดับต่อไปได้ ใช้แก้ปัญหา การขาดแคลนน้ำ และแหล่งเพาะปลูกที่มีอย่างจำกัด สภาพแวดล้อมที่ต้องการความควบคุมอย่างเหมาะสม กับการเพาะปลูกของพืชแต่ละชนิด

[7] จูติพงษ์ รัชชาริกรณ์ ได้วิจัยเรื่อง แนวทางการประยุกต์ใช้ Internet of Things (IoT) กับ Smart Agriculture 4.0 สาขาวิศวกรรมข้อมูลขนาดใหญ่ วิทยาลัยนวัตกรรมการเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตโดยงานวิจัยนี้ประยุกต์ใช้ Internet of Things ในการจัดการการปลูกพืชโดยใช้ เทคนิค Aquaponics (อควา โปนิกส์) ผ่านคอมพิวเตอร์ โทรศัพท์มือถือ หรือสมาร์ตโฟน โดยมีการเชื่อมต่อ กับอินเทอร์เน็ต ในที่นี้สามารถเชื่อมต่อได้ทั้ง แบบ LAN และ WIFI เพื่อเพิ่มความสะดวกในการใช้งานและ การเข้าถึงอุปกรณ์ต่างๆ ที่เราจำเป็นต้องสั่งการต่อไป ในการเชื่อมต่อนั้นจะผ่านระบบ IoT Cloud: Cayenne IoT Ready™ ซึ่งเป็น Server ให้บริการจัดเก็บข้อมูล รวมถึงการดูแลและการสั่งการต่างๆ ภายในระบบอีกที่ ทั้งนี้ในการทำงานส่วนใหญ่จะมีศูนย์ควบคุมหลัก คือ Raspberry Pi เป็นบอร์ดคอมพิวเตอร์ขนาดเล็กที่สามารถเชื่อมต่อกับจอมอนิเตอร์ คีย์บอร์ด และเมาส์ได้ สามารถนำมาประยุกต์ใช้ในการทำโครงการทางด้าน อิเล็กทรอนิกส์ การเขียนโปรแกรม หรือเป็นเครื่องคอมพิวเตอร์ตั้งโต๊ะขนาดเล็ก ไม่ว่าจะเป็นการทำงาน Spreadsheet Word Processing ท่องอินเทอร์เน็ต ส่งอีเมล หรือเล่นเกมส์ อีกทั้งยังสามารถเล่นไฟล์วีดีโอความ ละเอียดสูง (High-Definition) ได้อีกด้วย บอร์ด Raspberry Pi รองรับระบบปฏิบัติการลินุกซ์ (Linux Operating System) ได้หลายระบบ เช่น Raspbian (Debian) Pidora (Fedora) และ Arch Linux เป็นต้น โดย ติดตั้งบน SD Card บอร์ด Raspberry Pi นี้ถูกออกแบบมาให้มี CPU GPU และ RAM อยู่ภายในชิปเดียวกัน สามารถเขียนโปรแกรมเพื่อสั่งงานให้อุปกรณ์อิเล็กทรอนิกส์ทำงานได้ เช่น สั่งงานให้ Relay ทำงานตามเวลา ที่กำหนด หรือ สั่ง ปิด-เปิด ปั้มน้ำ ตามเวลาที่ต้องการ

[8] ดร.มหศักดิ์ เกตุฉ่ำ ได้วิจัยเรื่อง Internet of Things (IoT) ภาควิชาการจัดการเทคโนโลยี สารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือได้วิจัยประวัติ ที่มาและความสำคัญของอินเทอร์เน็ตสรรพสิ่ง(Iot)และหลักการทำงานรูปแบบต่างๆของอินเทอร์เน็ตสรรพ สิ่ง(Iot)

3. ขั้นตอนการดำเนินงานวิจัย

งานวิจัยนี้ได้ทำการวิจัยพัฒนาออกแบบเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานระหว่างแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor 2 โดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเทอร์เน็ตสรรพสิ่ง Internet of Things (IoT) โดยทดสอบผ่านเครื่องให้อาหารปลา โดยแบ่งออกเป็น 4 ส่วนการทำงานย่อย ได้แก่ ส่วนออกแบบแอปพลิเคชัน , ส่วนการรับส่งข้อมูล, ส่วนการบริหารจัดการผ่านโทรศัพท์เคลื่อนที่และ ส่วนทดสอบระบบเครื่องให้อาหารปลาสำหรับผู้ใช้งาน ตามภาพที่ 3.1



รูปที่ 3.1 ภาพรวมของระบบ

3.1 การออกแบบแอปพลิเคชัน Blynk

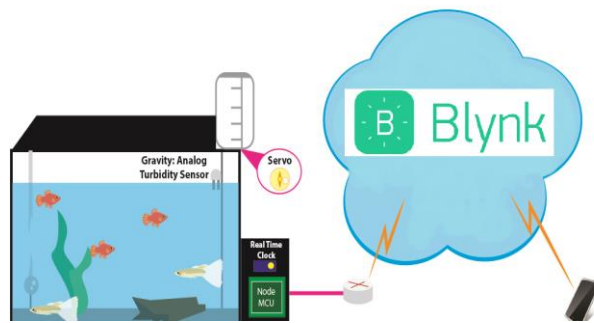
การออกแบบในส่วนแอปพลิเคชัน Blynk จะทำหน้าที่รับ-ส่งข้อมูลต่างๆและแสดงผลขึ้นทาง แอปพลิเคชัน โดยส่วนนี้จะให้บริการข้อมูลจากเซนเซอร์ผ่านกระดานข้อมูล(Dashboard) ในรูปแบบต่างๆ เช่น เวลาการให้อาหาร 3 ช่วงเวลาหรือการให้อาหารปลาอัตโนมัติ และค่าตัวเลขแสดงสถานะความขุ่นของน้ำในตู้ปลา เป็นต้น การออกแบบนั้นดูสวยงามและเข้าใจสถานะต่างๆ ได้ง่าย และสามารถเพิ่มเติมอุปกรณ์ต่างๆ บนกระดานข้อมูลได้ด้วยตนเองได้ และ แอปพลิเคชัน Blynk ยังมี Server และLibrary สำหรับอุปกรณ์ IoT เป็นแบบสำเร็จรูปจึงทำให้สะดวกต่อการเขียนโปรแกรมและลดขั้นตอนการเขียนโปรแกรมได้อีกด้วย แต่แอปพลิเคชัน Blynk นั้นจะมีในส่วนค่าใช้จ่ายเพิ่มเข้ามาในการซื้อENERGY BALANCE เพื่อที่ผู้ใช้จะสร้างไอคอนต่างๆ ของแอปพลิเคชัน Blynk โดยการซื้อ ENERGY BALANCE นั้นจะซื้อผ่านบัตรเครดิตหรือบัตรเครดิตแทน



รูปที่ 3.2 แสดง Appicition Blynk

3.1.2 การออกแบบระบบรับส่งข้อมูล แอปพลิเคชัน Blynk

ในการออกแบบผู้วิจัยได้ออกแบบให้ Blynk Server เป็นตัวกลางในการรับส่งข้อมูลโดยทำหน้าที่รับข้อมูลจากตัว Sensor Node แล้วส่งต่อไปยังที่ Blynk Server ต่อไป โดยระบบได้ถูกออกแบบมาให้ ตัว Sensor Node ทำการส่งข้อมูลมาเป็นระยะๆ ข้อมูลจะถูกส่งไปเก็บเอาไว้ที่ตัว Gateway แล้วจะถูกส่งต่อไปยัง Blynk Server ในส่วนฝั่งรับค่า ผู้วิจัยออกแบบให้รอรับค่าอยู่ตลอดเวลา เมื่อมีค่าร้องขอ ที่ถูกส่งเข้ามาใหม่ ระบบจะส่งไปยังตัว Sensor Node โดยทันที แล้วจะแจ้งสถานการณ์ปรับแต่งค่ากลับไปยัง ระบบ Blynk Server อีกครั้งหนึ่ง เพื่อเป็นการยืนยันการรับค่าสำเร็จ



รูปที่ 3.3 ระบบรับ-ส่งข้อมูลของแอปพลิเคชัน Blynk

3.1.3 การออกแบบ User Interface แอปพลิเคชัน Blynk

ในส่วนการออกแบบ User Interface ผู้วิจัยได้ออกแบบไว้เพื่อวัตถุประสงค์ให้ผู้ใช้งานสามารถใช้งานได้ง่ายและสะดวกต่อการใช้งาน ผู้วิจัยจึงได้ออกแบบในรูปแบบ แอปพลิเคชัน Blynk ที่สามารถเข้าถึงได้ง่ายจากโทรศัพท์เคลื่อนที่ และสามารถติดตามและสั่งการได้โดยง่าย โดยมีความสามารถสั่งการได้ดังนี้คือ

- ตั้งค่าเวลาในการให้อาหารปลาทองตามช่วงเวลา
- ตั้งค่าแสดงจำนวนปริมาณการให้อาหารปลา
- ตั้งค่าวัดระดับความขุ่นของน้ำในตู้ปลา
- ตั้งค่าในการให้อาหารปลาทองได้อัตโนมัติ

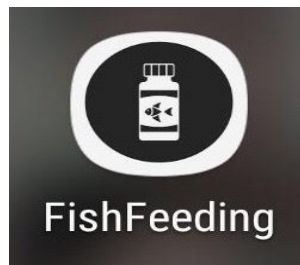


รูปที่ 3.4 แสดงข้อมูลการทำงานของ User Interface บนหน้าแอปพลิเคชัน Blynk

3.2 การออกแบบแอปพลิเคชัน Inventor 2

ในส่วนการออกแบบ แอปพลิเคชัน Inventor 2 ผู้วิจัยเลือกใช้ แอปพลิเคชัน Inventor 2 เพราะว่าเป็นแอปพลิเคชันที่สามารถออกแบบได้อย่างอิสระและไม่มีค่าใช้จ่ายใดๆ และการเขียนออกแบบแอปพลิเคชัน Inventor 2 นั้น โปรแกรมจะถูกออกแบบให้เขียนโปรแกรมผ่านทางหน้าเว็บไซต์โดยจะไม่มีตัวติดตั้งไฟล์โปรแกรมเหมือนโปรแกรมทั่วไปในการเข้าไปเขียนโปรแกรม แอปพลิเคชัน Inventor 2 นั้นจะต้องเข้าผ่านทาง Google Chrome และเข้าไปยัง www.google.com แล้วพิมพ์ค้นหาคำว่า แอปพลิเคชัน Inventor 2 จากนั้นเข้าไปในเว็บแอปพลิเคชัน Inventor 2 ในการจะเขียนนั้นต้องมี Gmail ในการ Login เข้าไปเขียนแอปพลิเคชัน Inventor 2 หลังจาก Login แล้ว ในการเขียน แอปพลิเคชัน Inventor 2 นั้น สิ่งสำคัญอีกอย่างคือ Internet เพราะจะต้องใช้ในการเชื่อมต่อในการออกแบบแอปพลิเคชันและในการ Login เข้าแอปพลิเคชัน Inventor 2 ทุกครั้งแอปพลิเคชัน Inventor 2 ที่ถูกเขียนขึ้นหรืออัปเดตนั้นก็จะถูกบันทึกเก็บไว้บน Cloud ของ แอปพลิเคชัน Inventor 2 ด้วย ในส่วนของหน้าของการออกแบบนั้นแบ่งได้เป็น 2 ส่วนคือ

ส่วนที่ 1 คือหน้า Designer เป็นหน้าที่ไว้สำหรับสามารถออกแบบไอคอนต่าง ๆ บน User Interface และส่วนที่ 2 คือหน้า Block เป็นหน้าที่ไว้สำหรับเขียน Code โปรแกรมในการสั่งการแอปพลิเคชันในการให้อาหารปลาหรือการแจ้งเตือนตั้งค่าความขุ่นของน้ำภายในตู้ปลา



รูปที่ 3.5 แสดง Application Inventor 2

3.2.1 การออกแบบระบบรับส่งแอปพลิเคชัน Inventor 2

ระบบการรับส่งข้อมูลนั้นแอปพลิเคชัน Inventor 2 นั้นจะต่างแอปพลิเคชัน Blynk ตรงที่จะไม่มี Server แบบสำเร็จรูปและ Cloud ของตัวเอง จึงจำเป็นต้องสร้าง Server และเขียนแอปพลิเคชันให้เชื่อมต่อกันและคำสั่งต่างๆขึ้นมาเอง จึงอาจจะมองว่ายากกว่า แอปพลิเคชัน Blynk ที่มี Server สำเร็จรูป และระบบทั้ง 2 ฝั่งเป็น Client อย่างเดียวจึงทำให้แอปพลิเคชัน Blynk ถูกมองว่าใช้งานและสร้างขึ้นง่ายกว่า และทำให้แอปพลิเคชัน Inventor 2 ยังมีข้อจำกัดการใช้งานอีกด้วยเพราะต้อง Forward Port ให้ระบบ Network ที่เชื่อมต่อสามารถมองเห็น Network จากภายนอกได้ ถ้าจะไปใช้ตามมหาวิทยาลัยหรือสถานที่สำคัญ ที่เป็นระบบ Network ที่ค่อนข้าง Security จะไม่สามารถใช้งานได้ แต่ข้อดีของ Server Application Inventor 2 จะมี

ข้อดีกว่า แอปพลิเคชัน Blynk ตรงที่เวลาระบบขัดข้องสามารถตรวจสอบแก้ไขได้ด้วยตนเองเพราะเป็นระบบ Server และ แอปพลิเคชัน ที่ผู้วิจัยเขียนขึ้นมาเองจึงง่ายต่อการดูแลตรวจสอบและพัฒนาต่อยอดได้อีกด้วย

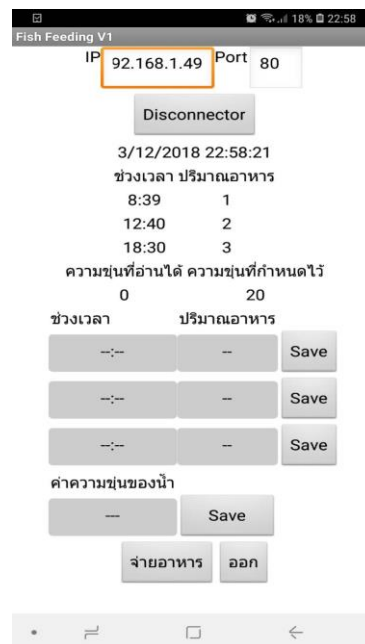


รูปที่ 3.6 แสดงระบบรับ-ส่งข้อมูลของแอปพลิเคชัน Inventor 2

3.2.2 User Interface แอปพลิเคชัน Inventor 2

การออกแบบส่วน User Interface ได้ออกแบบไว้เพื่อวัตถุประสงค์ให้ใช้งานได้ง่ายและสะดวกจึงได้ออกแบบในรูปแบบ Application ที่สามารถเข้าถึงได้ง่ายจากทั้ง โทรศัพท์เคลื่อนที่และคอมพิวเตอร์ผ่านหน้า Web browser โดยเน้นใช้งานที่โทรศัพท์เคลื่อนที่ที่สำคัญ สามารถติดตามและสั่งการได้โดยง่าย โดยมีความสามารถสั่งการได้ดังนี้คือ

- ตั้งค่าเวลาในการให้อาหารปลาทองในตามช่วงเวลา
- ตั้งค่าแสดงจำนวนปริมาณการให้อาหารปลา
- ตั้งค่าวัดระดับความขุ่นของน้ำในตู้ปลา
- ตั้งค่าเวลาในการให้อาหารปลาทองได้อัตโนมัติ



รูปที่ 3.7 แสดงข้อมูลการทำงานของ UserInterface บนหน้าแอปพลิเคชันInventor 2

4. ผลการดำเนินงาน

งานวิจัยนี้ผู้วิจัยได้นำแอปพลิเคชันสำเร็จรูประหว่างแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2 มาเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานในด้านต่างๆ โดยจะทดสอบผ่านทางพีทเจอร์และฟังก์ชัน และเงื่อนไขของแอปพลิเคชัน และจะทดสอบการใช้งานกับเครื่องให้อาหารปลาโดยผลการทดสอบจะอยู่ในตารางที่ผู้วิจัยทำขึ้นมาเพื่อให้เห็นการเปรียบเทียบข้อแตกต่างในด้านต่างๆ ของแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2 ว่าแตกต่างกันอย่างไรบ้างในด้านประสิทธิภาพด้านต่างๆ และสามารถนำไปใช้งานจริงได้ถูกต้องแม่นยำตามวัตถุประสงค์ของผู้ใช้งาน

4.1 ผลการทดสอบการเปรียบเทียบประสิทธิภาพการออกแบบหน้า User Interfaceและพีทเจอร์และฟังก์ชันเงื่อนไขต่างๆของแอปพลิเคชันBlynk กับ แอปพลิเคชัน Inventor 2

ผลการทดสอบจะเห็นได้ว่าแอปพลิเคชัน Blynk นั้นจะสามารถออกแบบในส่วนพีทเจอร์และในส่วนหน้า User Interface นั้นจะออกแบบได้ไม่เยอะเหมือนแอปพลิเคชัน Inventor2 แต่ในส่วนระบบการทำงานด้านการเชื่อมต่ออินเทอร์เน็ตและการใช้งานนั้นแอปพลิเคชัน Blynk จะสามารถใช้งานได้หลายระบบจะแตกต่างจาก แอปพลิเคชัน Inventor2ตรงที่สามารถใช้งานได้แค่ระบบ Android แต่ในส่วนฟังก์ชันต่างๆ การทดสอบทั้งสองแอปพลิเคชันจะไม่ต่างอะไรกันมาก จะแตกต่างตรงที่ว่าแอปพลิเคชัน Inventor2 ค่อนข้างจะเป็นฟังก์ชันที่ผู้ใช้สามารถแก้ไขออกแบบได้ด้วยตัวเองแต่แอปพลิเคชัน Blynk จะเป็นแอปพลิเคชันที่ค่อนข้างจะสำเร็จรูป ผู้ใช้จะไม่สามารถใส่ฟังก์ชันหรือออกแบบอะไรตามความพอใจ

4.2 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบแอปพลิเคชัน Blynk กับแอปพลิเคชัน Inventor2

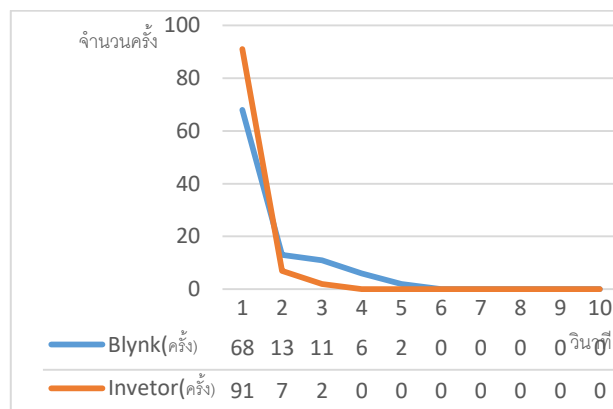
ผลการทดสอบจะเห็นได้ว่าแอปพลิเคชัน Blynk จะมีค่าใช้จ่ายในส่วนการออกแบบการสร้างแอปพลิเคชัน โดยผู้ใช้ต้องซื้อ ENEGRY BALANCE เพื่อที่จะสร้างไอคอนต่างๆในการสร้างแอปพลิเคชัน ซึ่งจะแตกต่างกับ แอปพลิเคชัน Inventor2 ที่จะไม่มียาใช้จ่ายใดๆทั้งสิ้น แต่แอปพลิเคชัน Inventor2 จะมีข้อจำกัดตรงที่ใช้งานได้แค่ระบบ Android อย่างเดียว

เปรียบเทียบประสิทธิภาพด้านงบประมาณค่าใช้จ่ายในการออกแบบและการนำไปใช้งาน	แอปพลิเคชัน Blynk	แอปพลิเคชัน Inventor2
ค่าใช้จ่ายในการออกแบบในส่วน User Interface	✓	✗
ค่าใช้จ่ายในการดาวน์โหลดในส่วนการนำไปใช้งาน	✓	✗

ตารางที่ 1.1 ตารางการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบแอปพลิเคชัน Blynk กับ
แอปพลิเคชัน Inventor2

4.3 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านเวลาในการทำงานของแอปพลิเคชัน Blynk กับ
แอปพลิเคชัน Inventor2

ผลการทดสอบประสิทธิภาพด้านเวลาในการทำงานของแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2 นั้น จะมีปัจจัยในการสั่งการใช้งานขึ้นอยู่กับอินเทอร์เน็ตที่ใช้ในการเชื่อมต่อในการรับส่งข้อมูล กรณีถ้าอินเทอร์เน็ตที่เชื่อมต่อนั้นมีความเสถียรปกติ แอปพลิเคชันนั้นจะสามารถสั่งการแอปพลิเคชันได้ทันที แต่ถ้าการเชื่อมต่ออินเทอร์เน็ตเกิดขัดข้องไม่ว่าจะเป็นไฟฟ้าดับหรือระบบอินเทอร์เน็ตขัดข้อง ก็อาจจะไม่สามารถสั่งการแอปพลิเคชันได้ทันที อาจเกิดการหน่วงด้านเวลาบ้าง โดยจากผลการทดลองผู้วิจัยสั่งทำการเชื่อมต่ออินเทอร์เน็ตทั้งสองแอปพลิเคชันแบบเดียว โดยจะจับเวลาในการสั่งการของแต่ละแอปพลิเคชัน โดยจะจับเวลาในการสั่งการ 100 ครั้งจะเห็นได้ว่าแอปพลิเคชัน Blynk นั้นจะเกิดการหน่วงเวลาในการรับส่งข้อมูลมากกว่าแอปพลิเคชัน Inventor2 เพราะว่าแอปพลิเคชัน Blynk นั้น Server อยู่บน Cloud ทำให้เวลาในการเดินทางข้อมูลอาจจะช้ากว่าแอปพลิเคชัน Inventor2 โดยผู้วิจัยได้ทำการทดสอบในการจับหน่วงเวลาโดยการสั่งการผ่านเครื่องให้อาหารปลาเป็นจำนวน 100 ครั้งของเวลาทุกวินาที



รูปที่ 3.8 รูปการเปรียบเทียบประสิทธิภาพด้านเวลาในการทำงานของแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2

4.4 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอปพลิเคชัน Blynk
กับ แอปพลิเคชัน Inventor2

ผลการทดสอบจะเห็นได้ว่าแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2 สามารถใช้งานได้ทั้ง Network Private IP และ Network Public IP แต่ในส่วนของแอปพลิเคชัน Inventor2 จะมีข้อกำหนดการใช้งานภายนอกตรงที่จะต้อง Forward Port และ Fix IP ของ Router เพื่อให้ Network Private IP สามารถ

มองเห็น Network ภายนอกได้ จึงจะสามารถใช้งานสั่งการผ่านแอปพลิเคชัน Inventor2 จากภายนอกได้ จะแตกต่างจาก แอปพลิเคชัน Blynk ที่มี Server อยู่บน Cloud จึงสามารถเชื่อมต่อ Network ที่ไหนก็ได้

เปรียบเทียบประสิทธิภาพด้าน การออกแบบการรับส่งข้อมูล	แอปพลิเคชัน Blynk	แอปพลิเคชัน Inventor2
การใช้งานผ่านเครือข่าย Network Private IP	✓	✓
การใช้งานผ่านเครือข่าย Network Public IP	✓	✓

ตารางที่ 1.2 ตารางการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2

5. สรุป

ในงานวิจัยนี้ได้เปรียบเทียบประสิทธิภาพการทำงานและการใช้งานงานระหว่างแอปพลิเคชัน Blynk กับแอปพลิเคชัน Inventor2 นั้น ซึ่งจะมีข้อแตกต่างกันทั้งในส่วนการออกแบบแอปพลิเคชันและในส่วนรูปแบบการรับส่งข้อมูลจากที่ผู้วิจัยได้ทำการเปรียบเทียบประสิทธิภาพในการทำงานและการใช้งานของทั้งสองแอปพลิเคชันนั้นจะสังเกตได้ว่าแอปพลิเคชัน Blynk จะใช้งานและเชื่อมต่ออินเทอร์เน็ตสะดวกและง่ายกว่าแอปพลิเคชัน Inventor2 เพราะมี Server สำเร็จรูปอยู่บน Cloud และในส่วนการเขียนโปรแกรมนั้นแอปพลิเคชัน Blynk ยังช่วยลดขั้นตอนการเขียนโปรแกรมได้อีกด้วยเพราะมีไฟล์ Library ที่สำเร็จรูปจึงทำให้ลดภาระการทำงานในส่วนของ Server และลดภาระทรัพยากรของคอมพิวเตอร์ไปอีกด้วยซึ่งจะแตกต่างกับแอปพลิเคชัน Inventor2 ที่ต้องเขียน Server ในการรับส่งข้อมูลขึ้นมาเองจึงมีข้อจำกัดในการเชื่อมต่ออินเทอร์เน็ตที่จะค่อนข้างยุ่งยากซับซ้อนกว่าแอปพลิเคชัน Blynk แต่ในส่วนของการออกแบบหน้าตาและฟิตเจอร์และฟังก์ชันต่างๆนั้น แอปพลิเคชัน Inventor2 จะสามารถออกแบบได้อิสระกว่าแอปพลิเคชัน Blynk ทั้งในส่วนหน้าตาของแอปพลิเคชันและในส่วนการสั่งการต่างๆและในส่วนด้านค่าใช้จ่ายในการออกแบบนั้นแอปพลิเคชัน Blynk จะมีค่าใช้จ่ายในการออกแบบแต่แอปพลิเคชัน Inventor2 จะไม่มีค่าใช้จ่ายแต่จะสามารถใช้งานได้ในระบบ Android อย่างเดียว แต่สามารถจำลองผ่านคอมพิวเตอร์ได้ และในส่วนแอปพลิเคชัน Blynk นั้นจะสามารถใช้งานได้ทั้งในระบบ IOS และในระบบ Android และเป็นที่ยอมรับ

แพร่หลายมากกว่าแอปพลิเคชัน Inventor2 ดังนั้นในงานวิจัยนี้ผู้วิจัยหวังว่าการเปรียบเทียบประสิทธิภาพการทำงานของทั้งสองแอปพลิเคชันนั้นจะเกิดประโยชน์สูงสุดต่อผู้พัฒนาและใช้งานต่อไป

5.1 สรุปผลการทดลองภาพรวมเปรียบเทียบประสิทธิภาพการทำงานด้านต่างๆระหว่างแอปพลิเคชัน Blynk กับ แอปพลิเคชัน Inventor2

1. แอปพลิเคชัน Blynk มีไฟล์ Library สำหรับอุปกรณ์ IoT ทำให้ช่วยลดขั้นตอนการเขียนโปรแกรม สามารถนำคำสั่งไปใช้งานในการรับส่งข้อมูลได้เลย แต่ในส่วนของแอปพลิเคชัน Inventor2 จะไม่มี Library ให้สำหรับอุปกรณ์ IOT จึงต้องเขียนโปรแกรม Server ในการรับส่งข้อมูลระหว่างอุปกรณ์กับแอปพลิเคชันขึ้นมาเอง

2. แอปพลิเคชัน Blynk มี Component ที่สามารถใช้งานได้สะดวก เพียงกำหนดค่า Parameter ให้ถูกต้องก็สามารถเชื่อมต่อกับอุปกรณ์ IOT ได้เลย ส่วน แอปพลิเคชัน Inventor2 ในการสั่งงาน Component จะต้องเขียนโค้ดโปรแกรมในการสั่งการขึ้นมาเอง

3. ระบบที่ใช้แอปพลิเคชัน Blynk อุปกรณ์ทั้ง 2 ฟังก์ชัน จะเป็น Client ทำให้ง่ายต่อการใช้งาน สามารถนำเครื่องไปใช้งานที่ใดก็ได้ ส่วนแอปพลิเคชัน Inventor2 ตัวเครื่องให้อาหารปลาจะทำหน้าที่เป็น Server ซึ่งการนำไปใช้งานจะต้องเซตค่า Network IP ให้ตัวกล่องสามารถมองเห็น Network จากภายนอกได้ ซึ่งมีข้อจำกัดในการนำไปใช้งาน

4. แอปพลิเคชัน Blynk สามารถใช้งานระบบโทรศัพท์เคลื่อนที่ได้ทั้ง 2 ระบบ Android และ IOS แต่ แอปพลิเคชัน Inventor 2 สามารถใช้ได้แค่ ระบบเดียวคือ ระบบ Android แต่สามารถจำลองผ่านคอมพิวเตอร์ได้

5. แอปพลิเคชัน Blynk จะมีข้อเสียคือ เป็น Sever ที่อยู่บน Cloud แบบสำเร็จรูปซึ่งเวลา Cloud หรือ Server มีปัญหาขัดข้องจะไม่สามารถตรวจสอบหรือแก้ไขอะไรได้เลยจะต่างกับแอปพลิเคชัน Inventor2 ที่มี Server เป็นของตัวเอง ซึ่งสามารถตรวจสอบแก้ไขได้ในกรณีที่ Server หรือระบบมีปัญหา

6. แอปพลิเคชัน Blynk จะมีค่าใช้จ่ายในการซื้อ ENERGY BALANCE เพื่อสร้างไอคอนต่างๆ แต่ในส่วนของแอปพลิเคชัน Inventor2 จะไม่มีค่าใช้จ่ายใดๆทั้งสิ้นในการสร้างไอคอนแอปพลิเคชัน

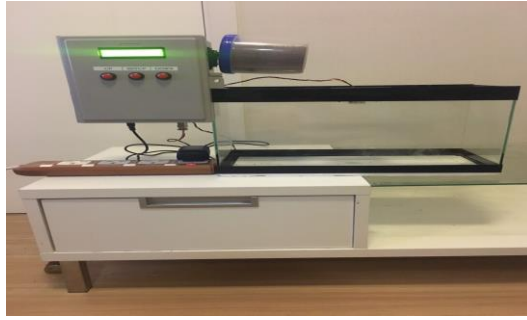
5.2 ข้อเสนอแนะแนวทางการพัฒนาต่อ

1. สามารถนำผลการทดลองนี้ไปเปรียบเทียบประสิทธิภาพในการทำงานและการใช้งานของแอปพลิเคชัน Blynk กับแอปพลิเคชัน Inventor2 ไปเปรียบเทียบประสิทธิภาพกับแอปพลิเคชันอื่นๆ ได้

2. สามารถนำความรู้เกี่ยวกับ แอปพลิเคชัน Blynk กับแอปพลิเคชัน Inventor2 ไปสร้างและพัฒนาต่อยอดโปรเจกต์อื่นๆได้ตามวัตถุประสงค์ที่ต้องการ

3. สามารถนำความรู้เกี่ยวกับแอปพลิเคชัน Blynk กับแอปพลิเคชัน Inventor2 นำไปพัฒนาต่อยอดเชิงธุรกิจต่อไปได้

4. สามารถนำความรู้ที่ได้รับเกี่ยวกับอุปกรณ์ (IoT) นำไปพัฒนาต่อยอดเพื่อให้เกิดประสิทธิภาพต่อไปได้



รูปที่ 3.9 แสดงเครื่องให้อาหารปลา

เอกสารอ้างอิง

- [1] การศึกษาค้นคว้าองค์ประกอบสำคัญในการเชื่อมต่อ Internet of Things เข้ากับ Cloud สืบค้นเมื่อ 4 มีนาคม 2562 , จาก <https://bit.ly/2FGOwMM>
- [2] การศึกษาค้นคว้า End-to-End Encryption สืบค้นเมื่อ 4 มีนาคม 2562 , จาก <https://bit.ly/2WH1BEJ>
- [3] สุชาดา พลาชัยภิมย์ศิลป์ (บทคัดย่อ : 2554) ได้วิจัยเรื่อง แนวโน้มการใช้แอปพลิเคชัน
- [4] พิเชิต บุญครอง* และ พิเชฏ บรมศรี ได้วิจัยเรื่องการพัฒนาแอปพลิเคชันเพื่อการพยากรณ์อนุกรมเวลาโดยใช้ MIT App Inventor MIT App Inventor-Based Application for Time Series Forecasting สาขาวิชาสารสนเทศการลงทุน วิทยาลัยเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยรังสิต
- [5] จามจุรี กุลยอด1 , ศิลป์ณรงค์ ฉวีพัฒน์2 ได้วิจัย เรื่องต้นแบบระบบควบคุมการเปิด-ปิดไฟผ่านแอปพลิเคชัน บนระบบปฏิบัติการแอนดรอยด์ Wireless Light Switch Prototype with an Android Application สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกำแพงเพชร
- [6] สุจินต์ ณ นคร ได้วิจัย เรื่องระบบควบคุมผักสลัดอัจฉริยะ (Smart Salad Planting) สาขาวิชาวิศวกรรมวิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์
- [8] ฐิติพงษ์ รักษาภิรมย์ ได้วิจัยเรื่อง แนวทางการประยุกต์ใช้ Internet of Things (IoT) กับ Smart Agriculture 4.0 สาขาวิศวกรรมข้อมูลขนาดใหญ่ วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์
- [9] ดร.มัทศักดิ์ เกตุหน้า ได้วิจัยเรื่อง Internet of Things (IoT) ภาควิชาการจัดการเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ